Ground Network and Range Software Release Notice

System to be Modified: ATS	**Release Number:**
Request for S/W Change Numbers[RSC(s)] : ATS-11 Meter CCO	**3.5**
Originator: Edward K. Payne **Phone:** (757)824-1104	**DRs Resolved:**
Date: 20 April, 2001	Wallops IDR 0414 and Wallops IDR 0362 (also known as CDS ID 00016248) NASA Lien List item #26

DR Short Title: ATS-11 Meter CCO

Design Review Required for RSCs: X Yes ☐ No **Date Design Review Held:** 07 February, 2001

Effort to Analyze RSCs: N/A Hours **Effort to Implement RSCs:** 1133 Hours

SLOC 35,000 total 400 added 200 changed 100 deleted

Date Change Required: 23 April 2001 **Change Required for Mission:** N/A

Modification Description – Provide General Description of Changes included in this Release (Include attachments to describe change/problem, if available)**:**

1) Added the display of text, which reflects current instrument status, to the high-level status windows on ATS Master Monitor & Control display for GDP911 Digital Matrix Switch and MSC 10693 Analog Matrix Switch.

2) Added ability to open and access an existing digital and/or analog matrix configuration file such that operations staff can quickly load and apply a new collection of switch settings of their choosing to its respective switch.

3) Added Master Monitor & Control modules that were modified to correct initialization problems uncovered through use of the "Bounds Checker" code integrity evaluation tool during Large Marge research at Goddard.

4) The Apogee 2208 Tracking Data Formatter (TDF) operations class was modified to handle "sweep in progress / sweep completed on SCC" status checking. SCC Sweep status was provided to enable checking of when to toggle doppler state on a TDF unit. Frequent failure of ATS on the node to determine the Apogee2208 recorded file name has been corrected.

Special Note 1: This delivery does NOT include any GDP911 Digital Matrix Switch redesign or code modification to incorporate the needs of the AQUA project, i.e. accommodate their desire for switch clearing. That effort is ongoing and will be included in a future release.

Special Note 2: The matrix switch interfaces provided in this delivery do not seamlessly "load" and "apply" the connections found in a switch configuration file. The "load" feature works for the first switch configuration file to be opened once the interface is started, however, second and subsequent opened switch configuration files will NOT load to the switches correctly. Users are restricted to using this feature ONCE per use of a switch interface. To re-enable the functioning of this feature, a user must completely exit the interface and restart it.

Special Note 3: The "Large Marge" font sizing aberration still exists in this release. It was determined during the appearance of Large Marge that font sizing grows from a 10 pt font to an 11 pt font in a different typeface. Although screen appearance may change slightly, Large Marge does NOT present an adverse functional impact to operational ATS computers.

Software Technical Lead Approval Signatures:	**Operation Technical Lead Approval Signatures:**
Date	**Date**

Software Build Instruction

Platform/Server Intel X-86 chipset CPU with Windows NT 4.0 / Service Pack 6.0 and MS Visual Studio 5.0 installed.

Input Directories (being delivered from):

See Build and Installation Instructions

Output Directories (being delivered to):

See Build and Installation Instructions

Build and Installation Instructions:

A few comments on minimum essentials

This part of the narrative will cover the minimum essentials that must be in place to successfully and correctly build the entirety of the ATS software environment.

You must have access to the complete ATS software developmental environment, either through:

- Remote access to the ATS source code archived on a Microsoft Visual Source Safe server, which is currently at ViaSat, Inc. facilities in Atlanta, Georgia, or
- Local access to the ATS source code by way of an ATS developmental environment archive CD that contains a certified copy of the entire, complete, ATS software environment, or
- Local access to the ATS source code by way of a network connection to a CSOC software engineer who has the complete ATS developmental environment on his workstation. Be advised that access through another workstation may be fraught with uncertainty, as there must be confidence established in the correctness of that individual's copy of the ATS developmental environment.

Moreover, you must have an Intel / Windows NT 4.0 – based workstation with:

- Microsoft Visual Studio / C++ 5.0 installed because as of 01 April, 2001, ATS does not use yet Microsoft Visual Studio / C++ 6.0 as its integrated development environment. (Plans are underway to modernize / upgrade to Studio 6.0)
- 700 MB of available drive space as there are a tremendous number of intermediate files that are created during ATS source code compiling / linking
- Compact Disk (CD) reader device to read the ATS development environment distribution CD
- (Optional) Compact Disk writer to create a run-time installation / distribution CD which will contain the installable executables and dynamic link libraries (DLLs) along with a number of other files discussed in this narrative.

A few comments on what's covered in the "ATS Build-Installation Instructions"

This part of the narrative introduces the overall process of what's involved in the creating of an ATS run-time / distribution CD. This process assumes that you have access to a CD writer and involves the following:

- The obtaining / retrieving the ATS development environment from the Visual Source Safe server or from a development environment archive distribution CD
- The building the ATS workspaces / projects in the correct sequence so as to ensure that libraries needed by later steps in the sequence are in place and available
- The creating of a "wrapper" directory (generally at the drive's root level) which will contain sub directories, which, in turn, will contain the executables, DLLs, and other files that are pulled from numerous places from inside the ATS development environment which will be written to a ATS release run-time / distribution CD
- The writing the wrapper directory (which can be considered "release candidate software"), to a blank CD

- The installing of the release candidate software from the run-time distribution CD to an ATS target computer

Working through the "everything" workspace world

This part of the narrative covers the first of the multiple workspaces / projects that must be touched to completely build the ATS environment. The first workspace is named "everything", which, admittedly, is somewhat of a misnomer, as there are additional workspaces that follow this one. The "everything" workspace contains all of the supporting elements that go into creating the general-purpose libraries, the individual instrument device drivers, and the general resource managers.

This section assumes that you have Microsoft Visual Studio 5.0 installed on your workstation and are familiar with basic concepts such as compiling and linking individual projects along with navigating around inside the IDE and the directory structure in which the ATS development environment exists.

- Start Visual Studio C++ 5.0 and open the workspace "everything.dsw", which should be found within the *: \ATS-Development\nodes\ats\project50. Note that "*:\" indicates the root level of the drive upon which you have placed the ATS development environment and that "ATS-Development" directory name may have additional characters which indicate the current or release candidate version number. The workspace consists of the following individual projects:
  - WGpp, WCom, WDev, BitSync, BSyncDecom7715, Demodulator, DemodAydin329A, Filter, FilterKrohnhite3905B, FrameSync, FSyncGdp225d, Matrix, MatHPE1366A, MatMSC10693, MatOptraxSS100B, MatOptraxSS3003S, Modulator, ModGDP783M, PCMSimulator, PCMSimGDP233, ProgTMProc, PTPAvtec1001, Recorder, RecMetrumBVLDS, Synthesizer, SynthHP3325A, SynthHP3325B, Test, TstWff123, Wff, WffTdf, MatGDP911, TDF, TDFApogee2208, ProcessManager, GRMPorts, RecMetrumBVLDS, MetrumBVLDSStatusDump, Wff123, GRM, and GRMMonitor
- Environmental Note: Observe that there is a building dependency chain, which starts at the beginning of the list and grows through the middle towards the end of the list. In the end the "GRMMonitor" project is dependent upon everything in front of it.
- Environmental Note: Observe that the outputs from the compiling and linking, be they "*.lib" files (which are used in subsequent, down stream projects at their respective compile-link time), "*.dll" files (which are used by resultant executables at run-time), or "*.exe" files (which are, in and of themselves, complete stand-alone applications), go to various locations within the ATS Development environment directory structure, depending upon their starting point in the dependency chain. These locations will be explained in the next section of this narrative.

Specific build steps within the "everything" workspace

This part of the narrative covers the individual steps that are needed to generate both a "debug" output and a "release" output version for each of the projects within the "everything" workspace. The reason for needing both is that some of the subsequent, down-stream, workspaces / projects have settings that anticipate that "debug" library files will be available, whereas other downstream workspaces / projects anticipate that the "release" library files will be available. Admittedly, this inconsistency may seem to be a problem, and will eventually be worked out, but for now, we build both types of output for the projects contained within the "everything" workspace.

To generate the "debug" type of build output, perform the following and check each box as you proceed through:

- Environmental Note: Ensure that the "output window" of the Studio IDE (generally located at the bottom) is made large enough to contain multiple lines of text. A window size of between 1/3 and ½ screen is good, as this will permit you to easily monitor the build progress.
- First, from the "Project" pull-down menu, select "Batch Build"
- Next, check the "debug" output of each of the project configurations
- Next, ensure that the "release" output of each of the project configurations is unchecked
- Next, select "Rebuild All" and note the output window for errors and warnings
- Next, observe that the build processing will create "debug" versions of the device dynamic link libraries (*.dll) and the compile / link time index libraries (*.lib) in the following directories:
  - "*:\Ats-Development\general\WLibs\DLLDebug50"

- "*:\Ats-Development\general\WDevs\DLLDebug50"
- "*:\Ats-Development\general\GRM\DLLDebug50"
- "*:\Ats-Development\general\MetrumStatusDump\DLLDebug50"
- "*:\Ats-Development\general\Wff123\DLLDebug50"

- Finally, observe that we will be interested in the content of the "WLibs", the "WDevs", and the "GRM" directories. The content of the other two debug directories will factor in at a later time

To generate the "release" type of build output, perform the following and check each box as you proceed through:

- First, from the "Project" pull-down menu, select "Batch Build"
- Next, check the "release" output of each of the project configurations
- Next, ensure that the "debug" output of each of the project configurations is unchecked
- Next, select "Rebuild All" and note the output window for errors and warnings
- Next, observe that the build processing will create "release" versions of the device dynamic link libraries (*.dll) and the compile / link time index libraries (*.lib) in the following directories:
  - "*:\Ats-Development\general\WLibs\DLLRelease50"
  - "*:\Ats-Development\general\WDevs\DLLRelease50"
  - "*:\Ats-Development\general\GRM\DLLRelease50"
  - "*:\Ats-Development\general\MetrumStatusDump\DLLRelease50"
  - "*:\Ats-Development\general\Wff123\DLLRelease50"
- Finally, as was the case with the "debug" output, observe that we will be interested in the content of the "WLibs", the "WDevs", and the "GRM" directories. The content of the other two release directories will factor in at a later time

Working through the "ATS" workspace world

This part of the narrative covers the second of the multiple workspaces / projects that must be touched to completely build the ATS environment. The second workspace is named "ATS" and contains all of the supporting elements that go into the creating of the "Operations" classes which are use by downstream, scheduled track event, automation processes.

- Start Visual Studio C++ 5.0 and open the workspace "ats.dsw" which should be found within the *: \ATS-Development\nodes\ats\project50. As was the case with the "everything" workspace, the "*:\" indicates the root level of the drive upon which you have placed the ATS development environment and that "ATS-Development" directory name may have additional characters which indicate the current or release candidate version number. The workspace consists of the following individual projects:

  - OpTsApogee2208, OpTsAvtec1001, OpTsAydin329A, OpTsDecom7715, OpTsGDP225D, OpTsGDP233, OpTsGDP783M, OpTsGDP911, OpTsHP3325B, OpTsHPE1366A, OpTsFilterKrohnhite3905B, OpTsMSC10693, OpTsMetrumBVLDS, OpTsOptraxSS100B, OpTsWff123, OpTsWffTdf, OpTsNodeManager, OpTrackingStation, GrmResources, TestOpTsNodeMgr.

- Environmental Note: Observe that there is a building dependency chain which starts at the beginning of the list, (which contains an "Operations" class for each of the instrument types) and grows towards the end of the list. Toward the end, you pass by the "General Tracking Station Operations" classes and eventually end up with the "TestOpTsNodeMgr" project, which is dependent upon everything in front of it.

Specific build steps within the "ats" workspace

This section covers the individual steps that are needed to generate both a "debug" output and a "release" output version for each of the projects within the "ATS" workspace. As was the case with the "everything" workspace, both types of compile / link output are needed and will be created here.

To generate the "debug" type of build output, perform the following and check each box as you proceed through:

- First, from the "Project" pull-down menu, select "Batch Build"
- Next, check the "debug" output of each of the project configurations
- Next, ensure that the "release" output of each of the project configurations is unchecked.
- Next, select "Rebuild All" and note the output window for errors and warnings.

- Finally, observe that the build processing will create "debug" versions of the device dynamic link libraries (*.dll) and the compile / link-time index libraries (*.lib) in the directory, "*:\ats-development\nodes\ats\DLLDebug50".

To generate the "release" type of build output, perform the following and check each box as you proceed through:

- First, from the "Project" pull-down menu, select "Batch Build"
- Next, check the "release" output of each of the project configurations
- Next, ensure that the "debug" output of each of the project configurations is unchecked.
- Next, select "Rebuild All" and note the output window for errors and warnings.
- Finally, observe that the build processing will create "release" versions of the device dynamic link libraries (*.dll) and the compile / link-time index libraries (*.lib) in the directory, "*:\Ats-Development\nodes\ats\DLLRelease50".

Working through the "DeviceGUIs" workspace world

This part of the narrative covers the third of the multiple workspaces / projects that must be touched to completely build the ATS environment. The third workspace is named "DeviceGUIs" and contains all of the supporting elements that go into the creating of the individual instrument interface windows applications.

- Environmental Note: The instrument interface application projects anticipate that supporting elements, such as "*.lib" files, to be in specific directory locations and that they are current. (Recall that these elements were created principally during the compiling and linking of the projects contained within the "everything" and "ats" workspaces earlier). If they are not present, nor current, then attempts at compiling and linking the instrument interface projects will produce uncertain results.

- Environmental Note: The projects within the DeviceGUIs workspace world have been set up to expect to see their needed supporting elements in the same directories that serve as destinations for the compile / link output from builds of both the "everything" and "ats" workspace worlds. It is presumed at DeviceGUIs workspace compile / link time that the desired library files are already in place.

- Procedural Note: If there have been no source code changes to modules within the "everything" and "ats" workspace worlds, then, most likely, the desired library files are ready for use by DeviceGUIs workspace projects.

- Procedural Note: If there have been changes to modules within the "everything" or the "ats" workspace worlds, then it is assumed that PRIOR to any attempt at building projects within the DeviceGUIs workspace world that:

  - the "everything" workspace will be totally rebuilt to make it current, following the steps found in the "everything" workspace section of this document

  - the "ats" workspace, which is, dependent upon the output of the "everything" workspace, will also be totally rebuilt to make it current, following the steps found in the "ats" workspace section of this document

- Start Visual Studio C++ 5.0 and open the workspace "DeviceGUIs.dsw", which should be found within the "*:\ATS-Development\Master\DeviceGUIs" directory. As was the case with the "everything" workspace, the "*:\" indicates the root level of the drive upon which you have placed the ATS development environment and that "ATS-Development" directory name may have additional characters which indicate the current or release candidate version number. The workspace consists of the following individual projects:

  - HP3325BSynthesizer50 – Win32 Debug
  - ModulatorGDP783 - Win32 Debug
  - OptraxSS3003SAnalogSwitch – Win32 Debug
  - PSKDemodulator - Win32 Debug
  - RemoteFilter – Win32 Debug
  - RemoteMetrum – Win32 Remote Metrum Interface
  - RemoteTDF – Win32 Debug
  - RFMatrixSwitch - Win32 Debug
  - RFSwitchEditor - Win32 Debug
  - Avtec1001PTP – Win32 Debug

- DigitalSwitch – Win32 Debug
- BitSynchronizer - Win32 Debug
- MatrixGDP911 – Win32 Debug
- MatrixMSC10693 - Win32 Debug
- RFSwitchEditor - Win32 Release
- RemoteMetrum – Win32 Remote Metrum Interface
- TdfApogee2208 - Win32 Release

- Environmental Note: Observe that some of the projects within the DeviceGUIs workspace produce a "Win32 Debug" version of output, whereas others produce a "Win32 Release" version of output, whereas, others, produce a generic "Win32" version of output. This is noteworthy in that it explains the necessity of producing both "debug" and "release" forms of the output libraries in the "everything" and "ats" workspace worlds, which are now being used at compile / link time within the projects contained within the DeviceGUIs workspace world.

- Environmental Note: The absence of a "Win32 Release" output being produced for each of the individual instrument interface projects is a holdover from the early days of ATS Master process developments. It is the goal of the ATS software sustaining team to, time permitting, re-architect the DeviceGUIs workspace world to produce just this type of resultant executable, that, in turn, exclusively uses "Release" forms of *.lib files during compile / link time and uses "Release" forms of *.dll files during run-time. For now, the afore-listed instrument interface windows application executable output types will be fielded.

- Environmental Note: Observe that, in contrast to the projects within the "everything" and "ats" workspace worlds, there is no formal dependency chain which starts off at the beginning of the list and grows towards the end of the list. Each of the individual instrument interface window application projects within the DeviceGUIs workspace stands on its own independently. The concept of the DeviceGUIs workspace is merely to provide a "convenience wrapper" around all of the individual instrument interface projects and in no way does any kind of dependency enforcement.

Specific build steps within the "DeviceGUIs" workspace

This section covers the individual steps that are needed to generate the respective output forms for each of the projects within the "DeviceGUIs" workspace.

To generate the "debug" type of build output, perform the following and check each box as you proceed through:

- First, from the "Project" pull-down menu, select "Batch Build"
- Next, check the "debug" output of each of the project configurations
- Next, ensure that the anything that is not a "debug" output (to include "release" and generic "Win32" output) of each of the project configurations is unchecked
- Next, select "Rebuild All" and note the output window for errors and warnings
- Finally, observe that the build processing will create "debug" versions of the instrument interface executables in the directory, "*:\Master". Observe the difference between the use of "..\..\..\DLLDebug50" directories within the "everything" and "ats" workspace worlds and the use of the "*:\Master" root-level directory, here

To generate the "release" and generic "Win32" types of build output, perform the following and check each box as you proceed through:

- First, from the "Project" pull-down menu, select "Batch Build"
- Next, check the output type box of each of the project configurations that isn't type "debug"
- Next, ensure that the "debug" output of each of the project configurations is unchecked
- Next, select "Rebuild All" and note the output window for errors and warnings
- Finally, observe that the build processing will create all of the output versions that are not of type "debug". For the DeviceGUIs workspace world these are "release" and generic "Win32" executables in the directory, "*:\Master"
- Observe the difference between the use of "..\..\..\DLLRelease50" directories within the "everything" and "ats" workspace worlds and the use of the "*:\Master" root-level directory, here

Following the compile / linking of each of the instrument interface projects within DeviceGUIs the help file for each device GUI should be "touched" to document any changes for the ATS release. This is easily done with the "Notepad" editor. The

device help files are located at "*: \Master and are named *devicename.txt*.

Working through the "Master-Development" workspace world

This part of the narrative covers the fourth of the multiple workspace / projects that must be touched to completely build the ATS environment. The fourth workspace is named "Master-Development" and contains all of the supporting elements that go into the creating of the individual applications that make up the collection of processes that handle the processing of scheduled, automated, track events, along with management of the overall ATS Master screen environment. The bulk of output from this section will go towards creation of the ATS Master "Monitor And Control" application, which is the centerpiece for ATS operation.

- Environmental Note: As was the case with the projects contained within the "DeviceGUIs" workspace world, the projects contained within the Master-Development workspace world anticipate that supporting elements, such as "*.lib" files, to be in specific directory locations and that they are current. If they are not present, nor current, then attempts at compiling and linking the Master-Development projects will produce uncertain results.

- Environmental Note: The projects within the Master-Development workspace world have been set up to expect to see their needed supporting elements in the same directories as did the projects within the DeviceGUIs workspace world. It is presumed at Master-Development workspace compile / link-time that the desired library files are already in place.

- Start Visual Studio C++ 5.0 and open the workspace "Master-Development.dsw", which should be found within the "*: \ATS-Development\Master" directory. The workspace consists of the following individual projects:

  - DSCInterface – Win32 Release
  - 11mInterface – Win32 Debug.
  - ManualNotification – Win32 Debug
  - MasterPassword - Win32 Debug
  - PassResultsCompiler - Win32 PassResultsCompiler
  - ProfileEditor – Win32 Debug
  - SAFSHeartbeat – Win32 Debug
  - Scheduler – Win32 Debug
  - ShippingReport – Win32 Debug
  - StationAssetsEditor – Win32 Debug
  - StationStatusDisplay - Win32 Debug
  - WOTISInterface – Win32 Debug
  - SAReceiver - Win32 Debug
  - RNInterface – Win32 Debug
  - SADSCNodeStatus – Win32 Debug
  - MonitorAndControl – Win32 Debug
  - PassResultsCompiler - Win32 PassResultsCompiler
  - RNInterface – Win32 Release
  - SAFSHeartbeat – Win32 Release

- Environmental Note: Observe, as was the case with the DeviceGUIs workspace world, that some of the projects within the Master-Development workspace produce a "Win32 Debug" version of output, whereas others produce a "Win32 Release" version of output, whereas, others, produce a generic "Win32" version of output.

- Environmental Note: As was the case with the DeviceGUIs workspace world, the absence of a "Win32 Release" output being produced for each of the Master-Development projects is a holdover from the early days of ATS Master process developments. In addition, as is the case with the DeviceGUIs workspace world, it is the goal of the ATS software sustaining team to, time permitting, re-architect the Master-Development workspace world to produce just this type of resultant executable. For now, the afore-listed scheduled, automation, event tracking, application executable output types will be fielded.

- Environmental Note: Observe, again, as was the case with the DeviceGUI's workspace world, there is no formal dependency chain which starts off at the beginning of the list and grows towards the end of the list. Each of the individual scheduled, automated, track event, application projects within the Master-Development workspace

stands on its own independently. The concept of the Master-Development workspace is merely to provide a "convenience wrapper" around all of the individual projects and in no way does any kind of dependency enforcement.

Specific build steps within the "Master-Development" workspace

This section covers the individual steps that are needed to generate the respective output forms for each of the projects within the "Master-Development" workspace.

To generate the "debug" type of build output, perform the following and check each box as you proceed through:

- ❑ First, from the "Project" pull-down menu, select "Batch Build"
- ❑ Next, check the "debug" output of each of the project configurations
- ❑ Next, ensure that the anything that is not a "debug" output (to include "release" and generic "Win32" output and any other) of each of the project configurations is unchecked
- ❑ Next, select "Rebuild All" and note the output window for errors and warnings
- ❑ Finally, observe that the build processing will create "debug" versions of the Master-Development executables in the directory, "*:\Master"

To generate the "release" and generic "Win32" types of build output, perform the following and check each box as you proceed through:

- ❑ First, from the "Project" pull-down menu, select "Batch Build"
- ❑ Next, check the "release" output type box of each of the project configurations that isn't exclusively type "debug"
- ❑ Do not select any output from any project configuration named "Unicode", or from "Release MinSize", nor from "Release MinDependency"
  - ❑ It is especially important that you only select generic "Win32" output and simple "Release" outputs!
  - ❑ Problems will result if you include any of the excluded three project configuration types!
- ❑ Next, ensure that the "debug" output of each of the project configurations is unchecked
- ❑ Next, select "Rebuild All" and note the output window for errors and warnings
- ❑ Finally, observe that the build processing will create the "release" or generic "Win32" version for each of the projects that have provided for one in the directory, "*:\Master"

Following the compile / linking of each of the instrument interface projects within Master-Development the help file for each of the resulting executable should be "touched" to document any changes for the ATS release. This is easily done with the "Notepad" editor. The help files are located at "*: \Master and are named the same as the individual projects within the Master-Development workspace. An example would be *MonitorAndControl.txt*. Note that not every project within the Master-Development has an associated help file.

Working through the "StartAnInterface" workspace world

This part of the narrative will cover of the fifth of the multiple workspaces / projects that must be touched to completely build the ATS environment. The fifth workspace is named "StartAnInterface" and contains the mechanics of how to invoke an instrument interface from a shortcut on a Remote Node computer. The resulting "StartAnInterface" application interprets the command arguments provided in the shortcut and directs the desired instrument interface windows application to start in the desired mode. It is essentially a stand-alone application.

- ❑ Start Visual Studio C++ 5.0 and open the workspace, "StartAnInterface.dsw", which should be found within the "*: \ATS-Development\General\StartAnInterface" directory. The workspace consists just one project: StartAnInterface, which has only one form of output, which is "Win32 – Debug"

Specific build steps within the "StartAnInterface" workspace

To generate the sole output version of "StartAnInterface" perform the following and check each box as you proceed through:

- ❑ First, from the "Project" pull-down menu, select "Batch Build"
- ❑ Next, select "Rebuild All" and note the output window for errors and warnings.

- ❑ Finally, observe that the build processing will create a "debug" version of the StartAnInterface executables in the directory, "*:\ATS-Development\General".

Preparation of the ATS Run-time Distribution CD Contents

This part of the narrative covers the creating of an ATS release candidate "wrapper" directory at the drive's root level, which will contain sub directories, which, in turn, will contain the executables, DLLs, and other files that are pulled from numerous places from inside the ATS development environment which will be written to an ATS release candidate run-time / distribution CD.

Distribution Wrapper Directory Prep

- ❑ Create a wrapper directory on your workstation. Check off each of the boxes as they are completed
  - ❑ The wrapper directory will be created at the drive's root directory level
  - ❑ The wrapper directory name should reflect the ATS release version and date as in ATS##-MMDDYYYY
  - ❑ An example: ATS34-12252000 describes the ATS release candidate applications built for release 3.4 on December 25, 2000
- ❑ Create four sub-folders. Check off each of the boxes as they are completed
  - ❑ "ATS##-MMDDYYYY \Master"
  - ❑ "ATS##-MMDDYYYY \Master\Station"
  - ❑ "ATS##-MMDDYYYY \Node"
  - ❑ "ATS##-MMDDYYYY \Wff"

Master Subdirectory Buildup

- ❑ Prepare the subdirectory named *Master.* Check off each of the boxes as they are completed
  - ❑ Use Windows Explorer to copy all of the executables (*.exe) and dynamic link libraries (*.dll) created within the "*:\Master" directory during the building of the "DeviceGUIs" workspace world and during the "Master-Development" workspace world into the "*:\ATS##-MMDDYYYY\Master" subdirectory
  - ❑ Use Windows Explorer to copy the following special executables and libraries from the listed locations into the "*:\ATS##-MMDDYYYY\Master" subdirectory.
    - ❑ The "*DSCInterface.exe*" executable from directory "*:\Ats-Development\Master"
    - ❑ The "*StartAnInterface.exe*" executable from directory "*:\Ats-Development\General"
    - ❑ The "*Grm.exe*" executable from directory "*:\Ats-Development\General\GRM\DLLRelease50"
    - ❑ The "*GrmResources.dll*" library from directory "*:\Ats-Development\nodes\ATS\DLLRelease50"
      - ❑ Special Note: Be aware that there are two forms of *GrmResources.dll*, the first of which is about 36-37kb in size and the second of which is around 200kb in size. We want a copy of the *GrmResources.dll* library which is around 200kb in size
    - ❑ The "*OpTrackingStation.dll*" library "*:\Ats-Development\nodes\ATS\DLLRelease50"
    - ❑ All of the "*OpTs*.dll*" libraries from directory "*:\Ats-Development\nodes\ATS\DLLRelease50"
  - ❑ Copy the file "*:\Master\Station*Initialization.txt*" into the Master\Station subdirectory. Edit the *Release* number at the bottom of the *Initialization.txt* file to reflect the current ATS release.
    - ❑ Special Note: There is the presumption that you had a working version of ATS Master code on your workstation.
    - ❑ Special Note: If not, contact the CSOC Software Sustaining Engineering group for the correct *Initialization.txt* file to place into your ATS run-time distribution CD prep directory.
  - ❑ Copy each of the ATS Master application "help" text files, along with each of the "hardware specs" text files, which should have been edited to reflect this release candidate, and are of type "*:\Ats-Development\Master*App.txt*" to the Master subdirectory

Node Subdirectory Buildup

- ❑ Prepare the subdirectory named *Node*. Check off each of the boxes as they are completed
  - ❑ Use Windows Explorer to copy all of the executables (*.exe) and dynamic link libraries (*.dll) created within the "*:\Master" directory during the building of ONLY the "DeviceGUIs" workspace world into the "*:\ATS##-MMDDYYYY\Node" subdirectory
    - ❑ Special Note: Be mindful of those files, which were created during the building of the "Master-Development" workspace world.
    - ❑ Special Note: "Master-Development" workspace output executables and libraries DO NOT get copied to the Node subdirectory!
  - ❑ Use Windows Explorer to copy the following special executables and libraries from the listed locations into the "*:\ATS##-MMDDYYYY\Node" subdirectory.
    - ❑ Special Note: These are the same special executables and libraries that were copied earlier into the "*:\ATS##-MMDDYYYY\Master" subdirectory
    - ❑ The "*StartAnInterface.exe*" executable from directory "*:\Ats-Development\General"
    - ❑ The "*Grm.exe*" executable from directory "*:\Ats-Development\General\GRM\DLLRelease50"
    - ❑ The "*GrmResources.dll*" library from directory "*:\Ats-Development\nodes\ATS\DLLRelease50"
    - ❑ The "*OpTrackingStation.dll*" library "*:\Ats-Development\nodes\ATS\DLLRelease50"
    - ❑ All of the "*OpTs*.dll*" libraries from directory "*:\Ats-Development\nodes\ATS\DLLRelease50"
  - ❑ Copy each of the ATS Instrument Interface application "help" text files and the "hardware specs" text files, which should have been edited to reflect this release candidate, and are of type "*:\Ats-Development\Node*App.txt*" to the Node subdirectory
    - ❑ Special Note: DO NOT copy the help text files associated with the ATS Master automation process applications, just those associated with the instrument interfaces.
    - ❑ Special Note: The names of these files will resemble the names of the projects found within the "DeviceGUIs" workspace world that was built earlier

Wff Subdirectory Buildup

- ❑ Prepare the subdirectory named *Wff*. Check off each of the boxes as they are completed
  - ❑ Use Windows Explorer to copy the following "test jig" executables that are used for diagnostic purposes into the "*:\ATS##-MMDDYYYY\Wff" subdirectory
    - ❑ The "*GrmMonitor.exe*" executable from directory "*:\Ats-Development\General\GRM\DLLRelease50"
    - ❑ The "*Wff123.exe*" executable from directory "*:\Ats-Development\General\Wff123\DLLRelease50"
    - ❑ The "*TestOpTsNodeMgr.exe*" executable from directory "*:\Ats-Development\nodes\ATS\DLLRelease50"
    - ❑ The "*MetrumBVLDSStatusDump.exe*" executable from directory "*:\Ats-Development\General\MetrumBVLDSStatusDump\DLLRelease50"
  - ❑ Use Windows Explorer to copy the following shared dynamic link libraries for the General Resource Manager into the "*:\ATS##-MMDDYYYY\Wff" subdirectory
    - ❑ The "*GRMRscController.dll*" library from directory "*:\Ats-Development\General\GRM\DLLRelease50"
    - ❑ The "*GRMRscManager.dll*" library from directory "*:\Ats-Development\General\GRM\DLLRelease50"
    - ❑ The "*GRMPorts.dll*" library from directory "*:\Ats-Development\General\GRM\DLLRelease50"
    - ❑ Special Note: The directory "*:\Ats-Development\General\GRM\DLLRelease50" contains a library named "GRMResources.dll", which is about 35-40 kb in size. This file has the same name as the library in the "*:\Ats-Development\nodes\ATS\DLLRelease50" folder, which is about 200 kb in size and was mentioned in both the "Master Buildup" and "Node Buildup" sections.
      - ❑ Do not copy this file as it does not have all of the information needed to run all of the ATS instruments

- Instead, simply keep the larger one copied in the "Master Buildup" and "Node Buildup" sections

- Use Windows Explorer to copy the following shared dynamic link libraries for the instrument device driver modules and supporting layers into the "*:\ATS##-MMDDYYYY\Wff" subdirectory

  - All of the "*.dll" libraries from directory "*:\Ats-Development\general\WLibs\DLLRelease50"
  - All of the "*.dll" libraries from directory "*:\Ats-Development\general\WDevs\DLLRelease50"
  - The "*Av_PTP.dll*" run-time from directory "*:\Ats-Development\General\Libraries\AvtecPTP"

- Use Windows Explorer to copy the following shared dynamic link libraries for the Visual Studio C++ run-time libraries and supporting elements into the "*:\ATS##-MMDDYYYY\Wff" subdirectory

  - All of the "*mfc*.dll*" **M**icrosoft **F**oundation **C**lass run-time libraries from directory "*:\Ats-Development\General\Libraries\Msvc"
  - All of the "*msvc*.dll*" **M**icro**S**oft **V**isual **C** run-time libraries from directory "*:\Ats-Development\General\Libraries\Msvc"
  - The "*Wininet.dll*" run-time library from directory "*:\Ats-Development\General\Libraries\Msvc"

Writing of the ATS Run-time Distribution Contents to Blank CD Media

This section covers the steps necessary to take the just prepared "wrapper" directory and copy it to a blank compact disk for distribution to the Ground Network operational sites.

For now, contact an ATS representative within the CSOC Software Sustaining Engineering group for the creation of copies of ATS run-time distribution CDs.

Staging and Activation of ATS Release Candidate Software

This section covers the steps necessary to successfully stage and activate new ATS release candidate software on an ATS target computer.

The steps presented here will guide the reader on how to

- How to create a "quarantine" directory for introduction of the new software to an operational ATS computer
- How to move the new "quarantined" software into "staging" directories in preparation for activation
- How to move the new software from the staging directories to the actual "run-time production" directories
- How to restore an operational ATS computer to a previous version of ATS software

- Procedural Note: The work of installing an ATS release candidate on a computer will be done inside of Windows NT Explorer.
- Procedural Note: Check off each of the boxes in the following sections as they are completed

Staging of ATS Release Candidate Software

- **Extra Special Procedural Note: Ensure that ALL ATS processes are shut down on ALL ATS computers** (to include each ATS Master and each ATS Remote Node computer) **PRIOR to beginning any ATS software staging work!** Follow these steps to gracefully bring down ATS:

  - First, shut down the ATS Master's MonitorAndControl process by way of the MonitorAndControl's "System" pull down menu "Shutdown M & C" entry. Supply the correct password when requested. Shutdown of the MonitorAndControl process will automatically shutdown the ATS Master's copy of a General Resource Manager (GRM)

  - Next, shutdown the "Operations" General Resource Manager (Ops GRM) on each of the ATS Remote Node computers by way of making the Ops GRM window the current window and then entering the "CTRL-C" keystroke combination to begin termination of this console application. Wait for the Ops GRM window to clear. You can observe activity within the slaved process "Device GRM" window during the Ops GRM window shutdown

- Finally, shutdown the "Device" General Resource Manager (Device GRM) on each of the ATS Remote Node computers by way of making the Device GRM window the current window and then entering the "CTRL-C" keystroke combination to begin termination of this console application. Wait for the Device GRM window to clear

- Environmental Note: The section, which covers the "copying / moving of a large quantity of files", will generally take much less time if old instrument and controller log files, stale schedule files, as well as PTP "metadata" and "flag files" are deleted from the ATS Master and Remote Node computers

  - ATS computer general housekeeping will greatly reduce the amount of data / number of files which must be manipulated on the disk, greatly speeding this process

  - Currently there is not a formal process for handling this, so ask one of the ATS Sustaining Engineers for assistance. A formal process document for performing ATS housekeeping is under development

Introduce the ATS Release Candidate Software To This Computer's Quarantine Area

- Start Windows NT Explorer on the ATS computer from which the ATS release candidate software will be installed

- Insert the ATS run-time distribution CD into the reader of this machine and attempt to expand the directory associated with it

  - Procedural Note: You may experience numerous read problems because of aging CD readers on the ATS computers
  - Procedural Note: Retry the attempt to read the CD not fewer than 10 times
  - Procedural Note: If the CD will still not read you may need to go to a different ATS computer with a CD reader that will accommodate the distribution CD

- Once the CD successfully is able to be read check to see if there is already a "C:\Downloads" directory present on this machine

  - If you don't have a "C:\Downloads" directory, create one. This directory serves as a system quarantine zone

    - First click on the "C:" drive icon in the directory tree window on the left portion of the Explorer display
    - Next, from the "File" pull down menu, select "New",
    - Next select "Folder",
    - Next enter the text "Downloads", then hit Enter

- Copy the wrapper directory "ATS##-MMDDYYYY" that you created in the section, "Distribution Wrapper Directory Prep" to the "C:\Downloads" directory on the current computer

Introduce the ATS Release Candidate Over to the other ATS Computer's Quarantine Areas

- Push the release candidate software to each of the ATS computers

  - Assign a drive letter to each of the other ATS computers in the tracking station
  - Ensure that you have a "C:\Downloads" directory on each of the other ATS computers within the station
  - Copy the wrapper directory "ATS##-MMDDYYYY" to the "C:\Downloads" directory on each of the other ATS computers

Create ATS Release Candidate Software 2nd Level Staging Environment

- In this subsection you will create the release candidate software "staging" environment on each of the ATS computers.

  - If you are on an ATS Master computer, create the following directories:
    - "C:\Master34" If this one already exists, delete it to include all of its contents, then create a new one
    - "C:\Master35"
    - "C:\Wff34" If this one already exists, delete it to include all of its contents, then create a new one
    - "C:\Wff35"

- Ensure that the "recycle bin" is empty to free up drive space
- Create this directory collection on each of the other ATS Master computers
- Complete the "Remote Node" staging directories setup that follows

- If you are on an ATS Remote Node computer, create the following directories:
  - "C:\Node34" If this one already exists, delete it to include all of its contents, then create a new one
  - "C:\Node35"
  - "C:\Wff34" If this one already exists, delete it to include all of its contents, then create a new one
  - "C:\Wff35"
  - Ensure that the "recycle bin" is empty to free up drive space
  - Create this directory collection on each of the other ATS Remote Node computers
  - Complete the "Master" staging directories setup just above

Create a Snapshot of the Current Operational State of each ATS Computer

- In this subsection you will do a large volume of file moving to duplicate the currently installed environment to enable the easy backpedal to the previous version of ATS software from the installation of the release candidate

  - For each of the ATS Master computers, copy, not move, the following directories and files:
    - The contents (to include all of the subdirectories of), but not the directory name, of "C:\Master" over to both the directories "C:\Master34" and "C:\Master35"
      - Procedural Note: The copy operation may take some time because of the multitude of mission support profile subdirectories that must be copied
    - The contents (to include all of the subdirectories of), but not the directory name, of "C:\Wff" over to both the directories "C:\Wff34" and "C:\Wff35"
    - Continue with the file moving in this subsection on each of the other ATS Master computers
    - Complete the "Remote Node" file moving directions that follow

  - For each of the ATS Node computers, copy, not move, the following directories and files:
    - The contents (to include all of the subdirectories of), but not the directory name, of "C:\Node" over to both the directories "C:\Node34" and "C:\Node35".
      - Procedural Note: The copy operation may take some time as all of the subdirectories are copied
    - The contents (to include all of the subdirectories of), but not the directory name, of "C:\Wff" over to both the directories "C:\Wff34" and "C:\Wff35"
    - Continue with the file moving in this subsection on each of the other ATS Master computers
    - Complete the "Master" file moving directions in the subsection just above

Move the ATS Release Candidate out of Quarantine Areas

- In this subsection you will move the release candidate files out of the quarantine directories over to the staging directories in preparation for formal activation in a later subsection

  - For each of the ATS Master computers, copy, not move, each of the following files from the directory "C:\Downloads\ATS##-MMDDYYYY\Master" over to "C:\Master35" directory. Overwrite each when prompted
    - Each of the executable files. "*.exe"
    - Each of the library files, ".dll"
    - Each of the text files, "*.txt"
    - The "*initialization.txt*" from the C:\Downloads\ATS##-MMDDYYYY\Master\Station directory to the "C:\Master35" directory
  - For each of the ATS Master computers, copy, not move, each of the following files from the directory "C:\Downloads\ATS##-MMDDYYYY\Wff" over to "C:\Wff35" directory. Overwrite each when prompted
    - Each of the executable files. "*.exe"
    - Each of library files, ".dll"
  - Complete the "Remote Node" file moving directions that follow

  - For each of the ATS Remote computers, copy, not move, each of the following files from the directory "C:\Downloads\ATS##-MMDDYYYY\Node" over to "C:\Node35" directory. Overwrite each when prompted
    - Each of the executable files. "*.exe"
    - Each of the library files, ".dll"

- Each of the text files, "*.txt"
- For each of the ATS Remote Node computers, copy, not move, each of the following files from the directory "C:\Downloads\ATS##-MMDDYYYY\Wff" over to "C:\Wff35" directory. Overwrite each when prompted
  - Each of the executable files. "*.exe"
  - Each of library files, ".dll"
- Continue with the file moving above in this subsection on each of the other ATS Master computers

Activation of ATS Release Candidate Software

- In this section, upon approval from the chief site shift lead, you will formally activate the release candidate software by moving it to the operations / production directories

  - For each of the ATS Master computers, copy, not move, each of the following files from the directory "C:\Master35" over to the "C:\Master" directory. Overwrite each when prompted
    - Each of the executable files. "*.exe"
    - Each of the library files, ".dll"
    - Each of the text files, "*.txt"
    - The "*initialization.txt*" from the directory "C:\Master35\Station" over to the "C:\Master\Station" directory
  - For each of the ATS Master computers, copy, not move, each of the following files from the directory "C:\Wff35" over to the "C:\Wff" directory. Overwrite each when prompted
    - Each of the executable files. "*.exe"
    - Each of library files, ".dll"
  - Complete the "Remote Node" file moving directions that follow

  - For each of the ATS Remote computers, copy, not move, each of the following files from the directory "C:\Node35" over to the "C:\Node"directory. Overwrite each when prompted
    - Each of the executable files. "*.exe"
    - Each of the library files, ".dll"
    - Each of the text files, "*.txt"
  - For each of the ATS Remote Node computers, copy, not move, each of the following files from the directory "C:\Wff35" over to the "C:\Wff" directory. Overwrite each when prompted
    - Each of the executable files. "*.exe"
    - Each of library files, ".dll"
  - Continue with the file moving above in this subsection on each of the other ATS Master computers

Reverting to the Previous Version of ATS Software

- In this section, upon approval from the chief site shift lead, you will "unload" the release candidate software and restore an earlier version of ATS software to the operations / production directories. Check off each of the steps as they are completed

  - For each of the ATS Master computers, copy, not move, each of the following files from the directory "C:\Master34" over to the "C:\Master" directory. Overwrite each when prompted
    - Each of the executable files. "*.exe"
    - Each of the library files, ".dll"
    - Each of the text files, "*.txt"
    - The "*initialization.txt*" from the directory "C:\Master34\Station" over to the "C:\Master\Station" directory
  - For each of the ATS Master computers, copy, not move, each of the following files from the directory "C:\Wff34" over to the "C:\Wff" directory. Overwrite each when prompted
    - Each of the executable files. "*.exe"
    - Each of library files, ".dll"
  - Complete the "Remote Node" file moving directions that follow

  - For each of the ATS Remote computers, copy, not move, each of the following files from the directory "C:\Node34" over to the "C:\Node"directory. Overwrite each when prompted
    - Each of the executable files. "*.exe"
    - Each of the library files, ".dll"
    - Each of the text files, "*.txt"

- ❑ For each of the ATS Remote Node computers, copy, not move, each of the following files from the directory "C:\Wff34" over to the "C:\Wff" directory. Overwrite each when prompted
  - ❑ Each of the executable files. "*.exe"
  - ❑ Each of library files, ".dll"
- ❑ Continue with the file moving above in this subsection on each of the other ATS Master computers

Special Instructions:

Delivered to CM by: Date

Build Completed by: Initial Release Candidate: Edward K. Payne Date: 20 April, 2001
Final Release to Operations: DeAndre' Hayward Date:

Testing Approved by: Karen Clark Date

Operations Approval to Install: James R. Hendrickson Date

Installation Completed by: Date

Software Installation Accepted by: Date